DRAFT

Preliminary TFM Information Architecture Steps

January 1998

Produced by:

Federal Aviation Administration
Architecture and Systems Engineering Division, ASD-100
NAS Information Architecture Team

Abstract

This paper describes an information analysis of Traffic Flow Management (TFM) systems that leads to an integrated view of and management of this information. The analysis describes a common data model for flight data to support TFM functions as well as those for free flight that can also be applied to other NAS domains.

KEYWORDS: air traffic control, information architecture, data model, data standard, system engineering repository, NAS, TFM, data management, data requirements, data architecture, metadata, flight data model

Table of Contents

List of Figures

List of Tables

Section 1

TFM Information Architecture Steps

The development of an information architecture is a subset of and supports the development of an overall system architecture. The purpose of an information architecture is to improve the efficiency and the quality of information delivery in the system. The information architecture identifies opportunities for improving the efficiency of information delivery including reduced life-cycle cost through shared development and maintenance of common information systems and services, the reduction or elimination of overlapping data sources, the elimination of duplicate information processing, and the creation of a cohesive methodology for sharing data. Improved information quality requires the development of consistent data definitions and standards, implementation of information security procedures and technology, and the identification of responsibilities for information quality. The development of an information architecture for the NAS is a major, long-term process as outlined in the main NAS Architecture, Version 3.0 documentation.

In parallel with the development of the information architecture concepts for the NAS Architecture, Version 3.0, the FAA has begun to develop some the more detailed components of an information architecture at both the NAS and the domain level. The following sub-sections describe initial steps in the development of an information architecture for systems supporting Traffic Flow Management (TFM). This work was initiated by the ATM IPT (AUA-500) and supported by ASD-110. The results reported are preliminary findings and are intended to provide a starting point for detailed coordination and development across the ATM IPT. Because of the breadth of TFM information requirements, these preliminary findings may be of interest to other domains and provide a starting point for the development of information architecture components in a NAS-level information architecture under the sponsorship of ASD-110.

Traffic Flow Management Data Analysis and Data Modeling. The development of an information architecture for the NAS is a major, long-term process, as outlined in the main NAS Architecture, Version 3.0 documentation. A much smaller effort in FY97 within the ATM IPT (AUA-500), with support from ASD-110 and CAASD, explored initial steps needed to develop an information architecture for the TFM operational environment. This section describes the processes established and the preliminary products generated with in this initial effort. In contrast to the top-down approaches used in the Concept of Operations-based flight information data model and the engineering repository just described, the work described below represents a bottoms-up approach that began with the analysis of the lowest level legacy data element and

ended up with the structuring of strawman standard data elements into the top-down data model.

Traffic Flow Management Common Data Working Group. The ATM IPT has recognized the need for all users of ATM information to have timely access to accurate, consistent, and reliable information. To that end, the ATM IPT chartered the TFM Common Data Working Group (TCDWG) to develop an architecture for TFM common data. By using the developed architecture to guide implementation and re-engineering, the ATM IPT will achieve:

- **Improved information access to NAS data for internal users and NAS users**
- **Standardized data and data services required to develop interoperable systems needed for information exchange and collaboration**
- **Improved data management and information access among ATM IPT applications**
- **Reduced data management costs (development, implementation, and operation) for individual systems.**

The TCDWG is chaired by AUA-500 with participation from ASD-100, MITRE/CAASD, and the AUA-TAC. Because the results of this effort will be directly applicable to the development of the NAS-wide Information System (NIS), as described in the NAS Architecture, the TCDWG will report its results to ASD's NAS Information Architecture Committee (NIAC).

Exploration Of Process. The TCDWG has established a series of steps to develop a Common Data Architecture for TFM systems and its future capabilities. Each of these steps provides a product or set of products that reflects the work performed in that step. Figure 1-1 illustrates these steps: identifying the data requirements, developing the data standard, developing a common data model, and providing a common data architecture.

The following sections describe the process used for completing each step and the progress made to date.

Identify Data Requirements. For the initial information architecture work described in this report, the TCDWG deliberately chose to limit the scope of its work to the analysis and structuring of the data that would appear as an input to or an output from a TFM application, system, program, or supporting system. The goal was to identify in as much detail as possible the data elements required or generated by each system. The level of detail available varied significantly. For existing operational systems very specific descriptions of the data elements used by the systems were generally available, but even in these cases the documentation was uneven, especially in

the definitions of the data elements. Descriptions of data elements required for research and development programs varied depending upon the maturity of the program. Data requirements for new, future systems as described in Concept of Operations documents and the TFM ART were least specific and often represented aggregated data components that could be decomposed into many distinct, lower level (atomic) data elements.

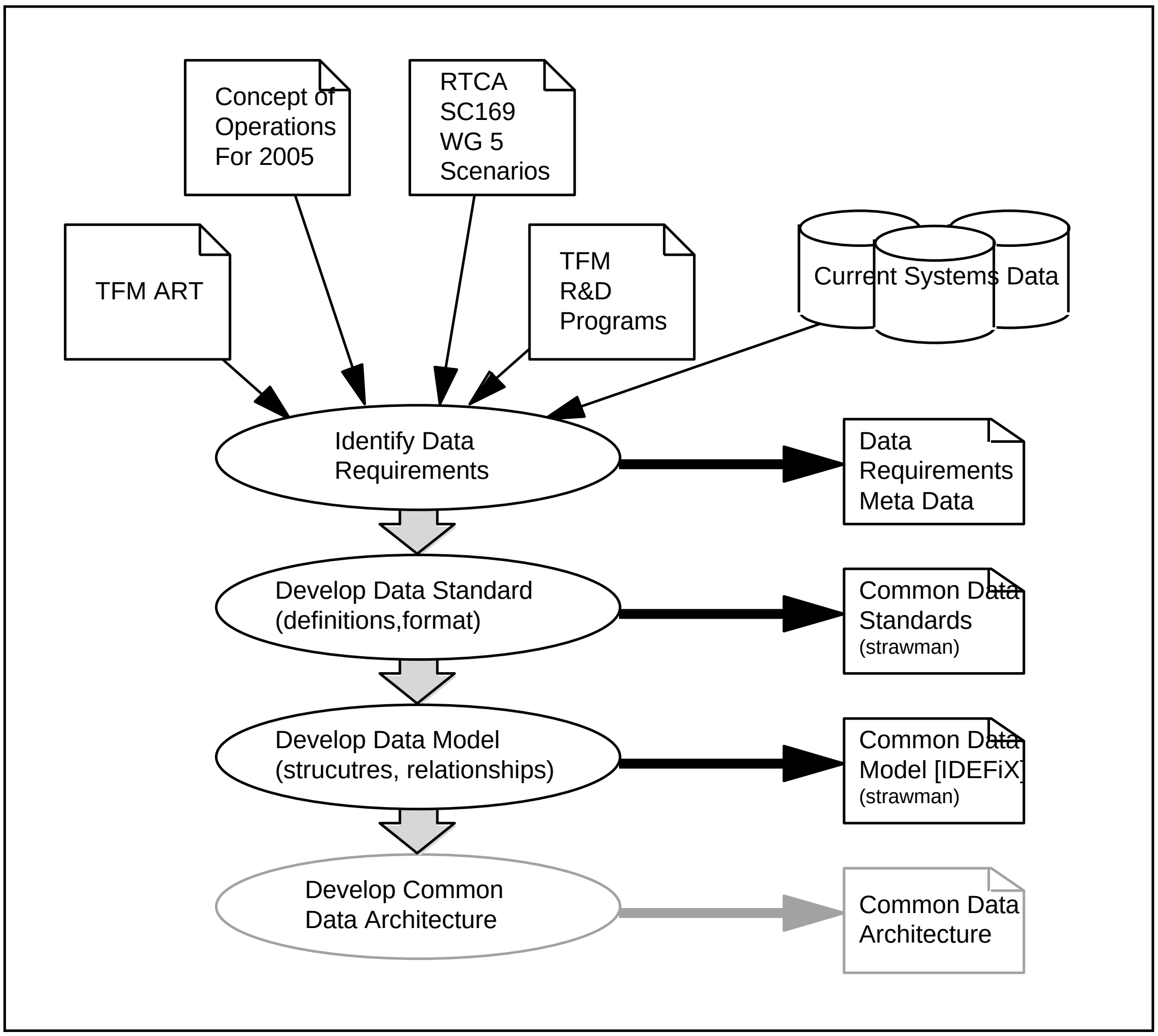

Figure 1-1. Steps to Develop a Common Data Architecture

Several TFM programs, including operational applications as well those currently in research and development, were analyzed to identify TFM data requirements. Only those data elements that are inputs to or outputs from the program were included in the inventory of elements, as the current focus is on shared data elements. Data elements were identified through existing documentation, including NAS-MDs, system specification and requirements documents, and other technical sources. Analysts from the AUA, AUA-TAC, and CAASD familiar with the programs and systems provided valuable assistance in identifying the data elements for this study.

Systems that were analyzed include CTAS (Center TRACON Automation System), TMA (Traffic Management Advisor), FSM (Flight Schedule Monitor)/CDM (Collaborative Decision Making), ETMS (Enhanced Traffic Management System), FAST (Final Approach Spacing Tool), Collaborative Routing, Fast-Time What If I, the Flight Object Model, HCS (Host Computer System), Interactive Flight Planning, OAG (Official Airline Guide), RTCA, and URET (User Request Evaluation Tool).

The TCDWG identified the data requirements using the following sources:

- **The NAS Concept of Operations[1] provides high level data requirements needed to support TFM operations through the year 2005.**
- **RTCA Special Committee (SC) 169 Working Group 5[2] AOC-ATM Information Exchange Scenarios were used to provide more detail and refinement of the high-level data requirements established from the NAS Concept of Operations. This source focused on the operations that involve information exchange with the user community. Additionally, the scenarios provide data requirements for a number of the future information exchange and collaboration capabilities identified in the AUA-500 R&D Plan.**
- **The results of the TFM-ART[3] were used to derive more detailed data requirements for current operations and future TFM capabilities. These data requirements serve to supplement the Concept of Operations data requirements.**

---

1 *A Concept of Operations for the National Airspace System in 2005, Revision 1.3,* Federal Aviation Administration, U.S. Department of Transportation, Washington, DC, June 1996.

2 *Operational Concepts and Information Elements Required to Improve AOC-ATM Ground-ground Information Exchange to Facilitate Collaborative Decision-making*, RTCA, April 29, 1997.

3 *The Future Traffic Flow Management System three volume set (Volume I: Operational Description, Volume II: Functional Decomposition, Volume III: System Architecture),* TFM Architecture Requirements Team (TFM-ART), 30 November 1993.

In addition, TFM-ART results provide the basis for a set of broad operational data categories to structure TFM data requirements collected from various sources. There are seven high level categories: Flight (demand), Resources (capacity), Weather, Traffic Management, General Resources, Performance, and Miscellaneous. Each high level category has a number of sub-categories. The general categories and sub-categories are described in more detail later in this document.

- **Current operational systems, operational prototype, and programs under development were reviewed to identify data requirements. The following were systems were examined: Enhanced Traffic Management System (ETMS), Center TRACON Automation System (CTAS) Traffic Management Advisor (TMA) and Final Approach Spacing Tool (FAST), Surface Movement Advisor (SMA), Dynamic Oceanic Tracking System (DOTS), Notice to Airmen (NOTAM), Airspace Analysis Tool (AAT), Aeronautical Information System (AIS), Flight Schedule Monitor (FSM). The inputs and outputs of these applications were documented and grouped into the broad categories mentioned above.**
- **Research and Development (R&D) programs (including specific programs under the following categories) are: Information Exchange, Ground Delay Program (GDP Enhancements), Collaboration Tools, and NAS Analysis and Prediction Tools. The R&D programs were examined for data requirements. These data requirements were assigned to a category mentioned above. It should be recognized that many of the R&D programs are in very early stages of Concept Exploration and Development and do not have well-defined data requirements.**

Several items of metadata were collected for each legacy element, as illustrated in Figure 1-2. These included system element name, its definition, its description, and its data type and format. In addition, it was noted whether the data element was an input or output, and what the source and destination system(s) were. Where possible, information about the complexity of the data and its internal system name and structure was captured. Finally, the data element was assigned a common data category.

A standard element name(s) to which a data element would be mapped was assigned after a standard element was identified. This was done after the elements from all of the legacy systems were categorized and grouped together by data category.

Common data categories were developed to provide a framework for organizing the legacy data elements. By organizing the data in this fashion, the TCDWG was able to partition the data elements identified into more manageable subsets for identifying and defining standard data elements by logical groupings. These categories are not final and may changed as analysis of data analysis proceeds.

There are currently seven major data categories as shown in Table 1-1, each of which has several sub-categories. The first category is flight, or demand data. This includes information such as the flight itinerary, flight identification, flight planning, flight events and status, and ATM control events that affect a single flight. Second is resource, or demand data describing static resources, such as airports, runways, and airways, as well as their dynamic status, such as configuration or activation. Third is weather data, which includes terminal and airborne weather observations, forecasts, and reports of weather phenomena. Fourth is traffic

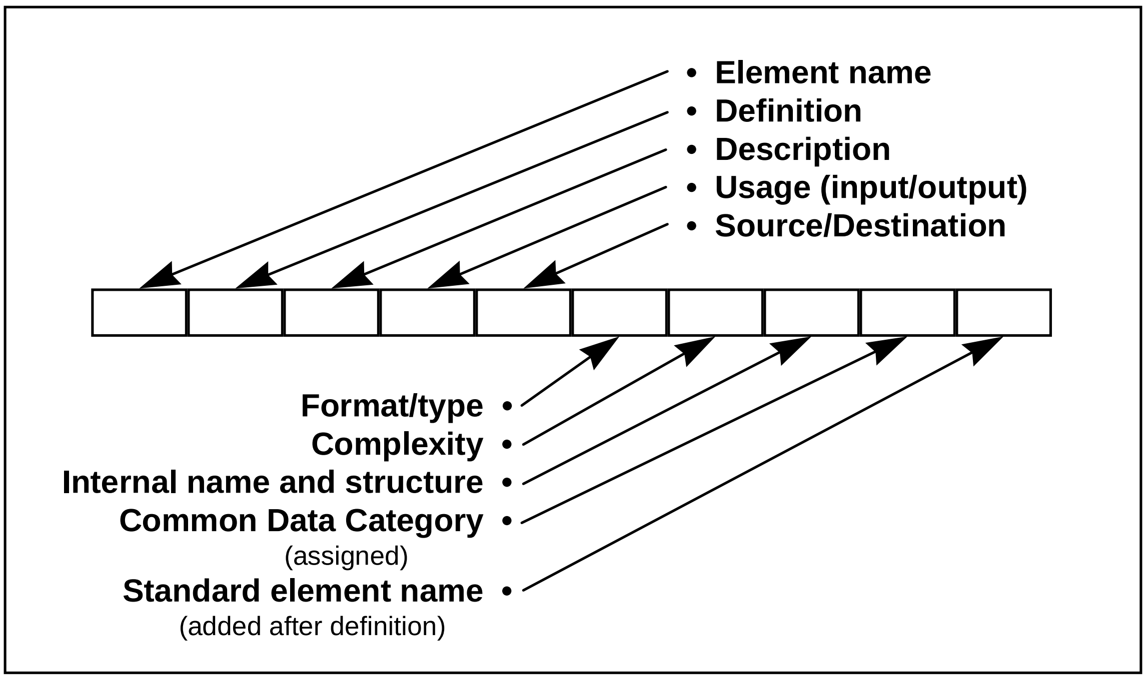

Figure 1-2. Metadata for Identified Data Elements

management data describing the situations in which capacity exceeds resources, and actions taken by ATC, TFM, and users to resolve these imbalances. Fifth is general resources data that are not NAS-specific, such as time, geography, and geopolitical data. Sixth is performance data that is used to describe NAS operational effectiveness and its ability to meet user needs. The final category is a catch-all for miscellaneous data elements that do not fit under the other major categories. Conceivably, new major categories could be identified and defined at some later time.

The data requirements from all sources and associated categories is available in an Excel spreadsheet.

Develop a Data Standard. The TFM data requirements provide the basis for the development of the TFM data standard which is a proposed authoritative representation of the common data that includes a naming convention, format, validation rules, business rules and definitions. Other applicable standards (e.g., ICAO flight plan and standards defined in the TFM Domain Definition Document) will also be considered in establishing the data standard. The data standard will seek to provide a common definition for all characteristics (e.g., format) of the data elements in the set of data requirements. The steps used to develop the common data elements (standard) are illustrated in Figure 1-3. To provide an understanding

Table 1-1. Data Categories Used to Organize TFM Data Elements

	High-Level Data Category	Definition
1.	Flight (Demand)	The data required by NAS users and operators to describe, manage, and control the safe movement of aircraft in the NAS. Much of this data is associated with one flight where a flight normally includes one take-off and one landing.
2.	Resource (Capacity)	The description (static and dynamic) of NAS resources and the management of these resources
3.	Weather	Data describing forecast and observed weather conditions and phenomena
4.	Traffic Management	The data describing actions taken by users and the FAA (ATC and TFM) that affect how collections of flights are planned, flown, or predicted.
5.	General Resources	The description or status of non-NAS that affect flights as planned, flown, or predicted.
6.	Performance	Data, metrics, and models used to measure FAA operational effectiveness and responsiveness to user needs
7.	Miscellaneous	Miscellaneous data

of how the TFM data standard was developed, a mapping will be produced that relates the elements in the data standard to the data requirements. The mapping will provide traceability of the sources of the requirements to the defined standard.

For each TFM related system, the metadata for its data elements were compiled in an Excel spreadsheet. A common data category was assigned and included in the metadata, which were then sorted by common data category. At this point, all items from a single high-level category could be extracted into a separate spreadsheet.

Data elements from multiple TFM systems that appeared to represent the same information were grouped together, and a standard element name was assigned to all of the data elements in the group. Subsequently, standard element definitions, data types, and units and valid values were developed, based on the legacy elements, or on other identified standards. Although data exchange requirements collected include data elements in all data categories, the initial focus for the analysis was restricted to the Flight (Demand, or category 1).

A number of steps taken to establish a set of standard data are shown in Figure 1-4. First, a common data category was selected; in this case, category 1 was chosen. Some of the legacy data elements were complex and had to be decomposed into simple, atomic data elements. For example, 'position' data split into latitude, longitude, altitude and time components. Next, as noted previously, 'like' elements that seemed to identify the same data item were grouped together. A standard element name was assigned, based on naming

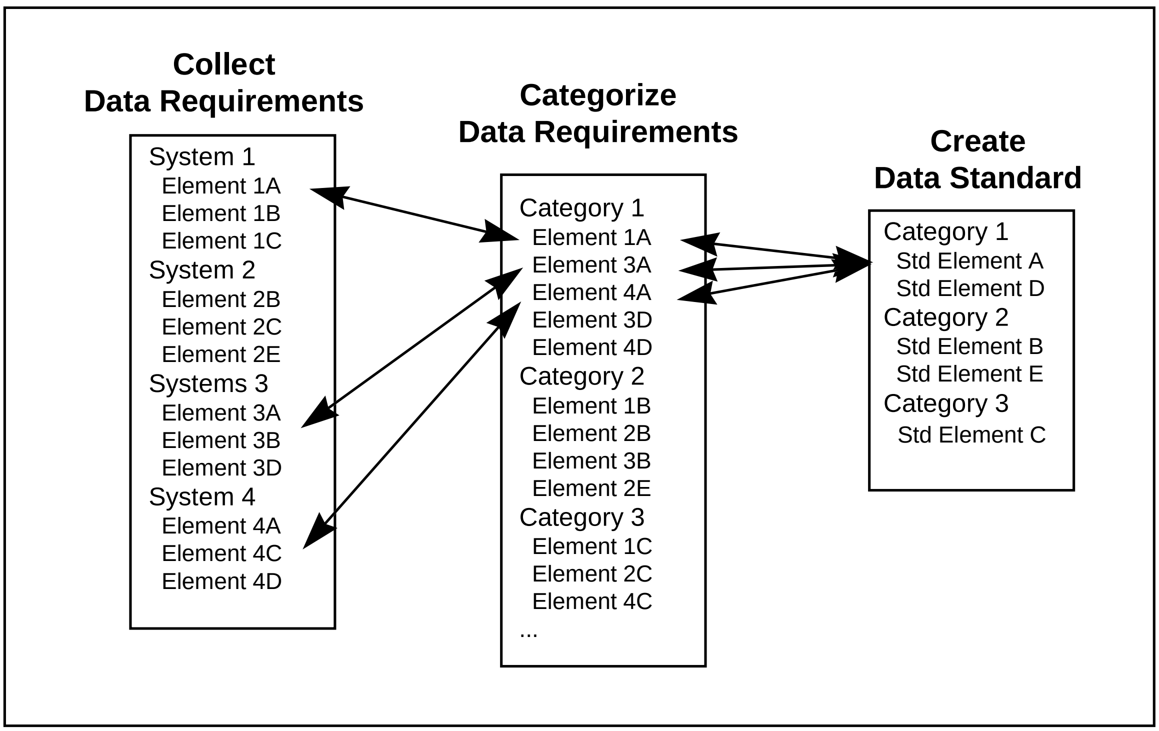

Figure 1-3. Steps to Define a Common Standard Data Element

conventions determined previously. A definition was developed, either based on one of the legacy element definitions, documents such as NAS-MD-311, 7110.65, or other standards. The standard data element was then further described by assigning a data type, a format, and valid values if known. Finally, the standard data elements were mapped to the legacy elements, and vice-versa.

Metadata defined for the standard data elements shown in Figure 1-5 include element name (based on an established naming convention), element definition, data type and format; units of measure, and valid values, if known.

The initial version of the data standard generated by the TCDWG will be made available for review by the TFM community. When agreed to and adopted by the TFM community, the TFM data standard will serve as an authoritative standard for future new development and re-engineering of current operational programs.

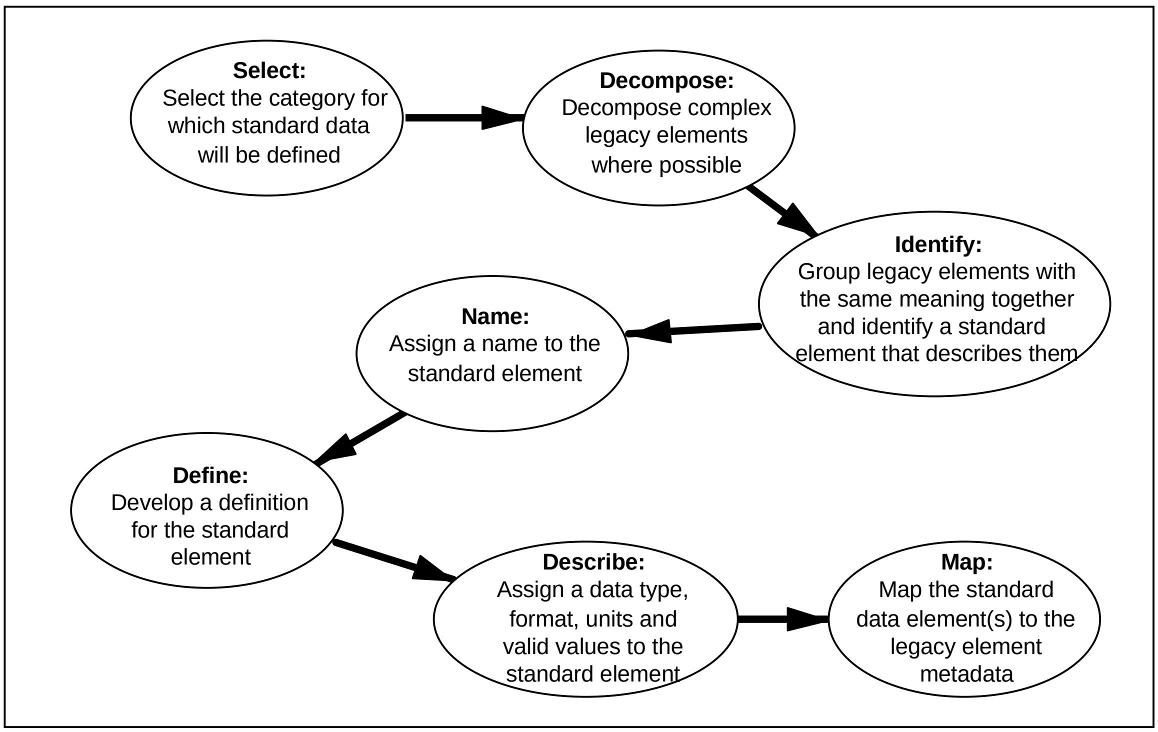

Figure 1-4. Steps for Establishing Standard Data

Microsoft's Access database product was used to manage the legacy and standard data. Two tables, corresponding to the spreadsheets created for legacy and standard

data, were created and joined by the standard data name. The spreadsheets were imported into the tables, and traceability between the legacy and standard elements was enabled via a standard data name.

Putting system metadata into a relational database offers several advantages. First, as mentioned previously, two-way traceability between the standard and legacy data elements is possible. Second, other traceability, such as between the standard elements and the legacy *systems*, is also possible. Third, queries can be used to extract information about relationships between the standard and legacy data elements: Do all the legacy elements have a corresponding standard data element? Queries can also be used to query about a specific type of data; for example, to identify all legacy elements relating to flight plan amendments. Finally, changes can be made to standard data element metadata without affecting its relationship to the standard data. For example, if a standard element name is changed, there is no need to change it for every affected legacy element's metadata. Instead, the change is propagated to all the corresponding legacy elements.

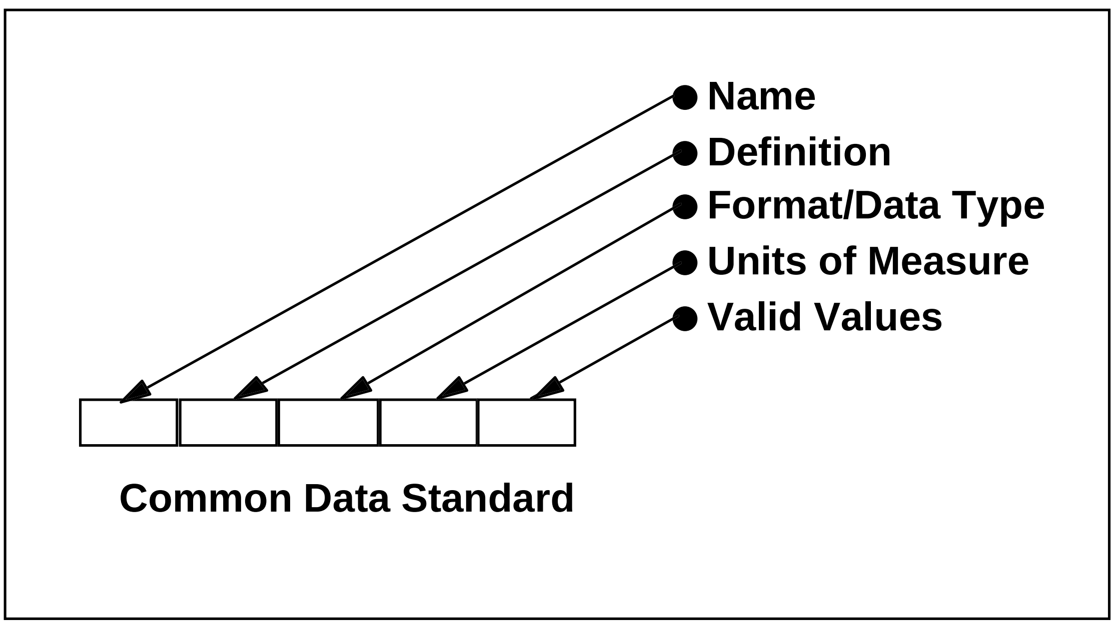

Figure 1-5. Metadata for Common Data Standard

Numerous contexts were found for some TFM data elements. For example, an *arrival time* value could be 'preferred,' 'filed,' 'assigned,' or 'actual.' The contextual descriptors shown in Table 1-2 found in the legacy data included: 'actual,'

‘assigned,’‘calculated,’ ‘earliest,’ ‘estimated,’ ‘filed,’ ‘next,’ ‘planned,’ ‘predicted,’ ‘previous,’ ‘projected,’ ‘requested,’ and ‘scheduled.’ Some of these terms, such as ‘calculated/estimated’ and ‘projected/predicted’ appear to be redundant, or have contradictory meanings from one usage to the next.

A set of context ‘types’ was selected and defined. For every standard data element that has multiple contexts, a corresponding ‘type’ element was added. For example, the data element arrival_time_runway is paired with an element called arrival_time_runway_type, which gives the context for the arrival runway; that is, ‘actual’, ‘assigned’, ‘filed’, or preferred’. Developers should be able to select the ‘type’ variable that is appropriate for the context they are defining.

Table 1-2. Contextual Data Descriptors

Selected Word	Meaning	Example
acceptable	User-specified resource or bound on event time willingly received by the user.	arrival_runway_delay_acceptable
actual	Event time or resource used, known definitively only once the event has occurred or the resource has been used.	departure_route_actual
assigned	Event time or resource designated by ATC.	arrival_position_assigned
earliest	User-specified lower bound on time when an event occur can occur.	departure_time_runway_earliest
filed	Refers to the flight plan as filed with an ATS unit by the pilot or his designated representative without any subsequent changes or clearances.	diversion_airport_filed
modeled	Refers to characteristics of an aircraft that are used to define its trajectory.	aircraft_altitude_rate_modelled
original	Refers to event times of controlled flights prior to changes due to ground-delay programs	departure_time_gate_controlled_original
planned	Anticipated resource or event – user view	arrival_time_runway_planned
predefined	Static NAS or ATC resources, such as ATC Preferred Routes, whose definitions are published.	route_predefined
preferred	Refers to static user preference established prior to flight filing.	departure_taxiway_preferred
previous	Event/state that occurred immediately prior to the current event/state.	position_latitude_previous
projected	Event/state to occur in future whose actual time/location are empirically, bu not definitively, known. Synonymous with: calculated, estimated, next, predicted, expected.	arrival_time_runway_projected
proposed	Anticipated [inactive] filed flight plan event – ATC view	arrival_time_runway_proposed
requested	Indicates a dynamic user preference during flight execution; e.g., change to another altitude due to turbulence.	altitude_cruise_requested
scheduled	Refers to published OAG or to in-flight events whose occurrence are desired at a particular time and/or location.[should we split into published and scheduled?]	arrival_time_gate_scheduled arrival_time_crossing_point_scheduled (desired time that an aircraft should cross a specific point)

Develop a Common Data Model For Flight Data. For the initial analysis, the TCDWG focused on data pertaining to flights; that is, flight schedules, flight plans, flight progress, and related aircraft data. The Concept of Operations for the NAS in 2005 identifies a flight data thread in the NAS-wide information system, providing "information on each flight from the moment of push-back to wheels-up, including surveillance data in flight, touchdown time and gate assignment." The initial version of the common flight data model is described in detail in Section 2.

Provide a Common Data Architecture. In this step, the TCDWG planned to compile the work done in previous steps to identify future activities that should be performed to assist the ATM IPT in implementing the data standard and driving toward common data. This work is in progress. Results from this step will be:

- **Guidance, principles, and transition strategies for the management of the data**
- **Description of data administration functions**
- **A process for maintaining and modifying the architecture**
- **Identification of suggested follow-on activities**
- **Summary of the impact of this work on the TFM technical architecture.**

These products will provide the IPT with possible approaches and activities to follow that would lead to the implementation of the common data architecture in the TFM domain.

Section 2

Draft Flight Data Model

The Air Traffic Service (ATS) Concept of Operations for the NAS in 2005 identifies a flight data thread in the NAS-wide information system, providing "information on each flight from the moment of push-back to wheels-up, including surveillance data in flight, touchdown time and gate assignment."[4] The Concept of Operation envisions that this information on each flight will be directly accessible and consistent across all users.

Many NAS applications use flight information. Today, each uses its own view of this information for a number of reasons. Increasingly, to implement the Operational Concept and to build systems faster and cheaper and with greater interoperability, NAS systems should be built with commonly accepted data structures. This does not prevent those systems from customizing the use of these data or the view of these data for application-specific needs. However, common data structures will make many information management tasks easier, such as data exchange and application maintenance. The data model described in this document is an initial version of a structured way to represent flight data that can support a wide range of users and applications.

Although development of the flight data model has been in direct support of TFM, the design itself is generic and is applicable to any system requiring flight data. Most of these data, such as flight schedules, flight intent, aircraft and operator descriptions, are generated by the air carriers. Other demand data, such as data about flights in progress and forecast flight data, are generated by the ATM system.

Flight data need to be shared throughout the NAS. For ATC and TFM, they need to be shared and accessed in real and near-real time by both FAA service providers and Traffic Management Coordinators (TMCs) but also by Aeronautical Operational Control (AOCs) and pilots. System developers need to know the structure and location of flight data if the systems uses or produces such data.. The Operational Concept describes how a common flight data structure will support current and future operational procedures and requirements. The preliminary flight data model presented here All of these needs can be accommodated by.

---

4 *A Concept of Operations for the National Airspace System in 2005, Revision 1.3,* Federal Aviation Administration, U.S. Department of Transportation, Washington, DC, June 1996.

Figure 2-1 shows how the demand data modeled by the flight data model is used in a phase of flight.

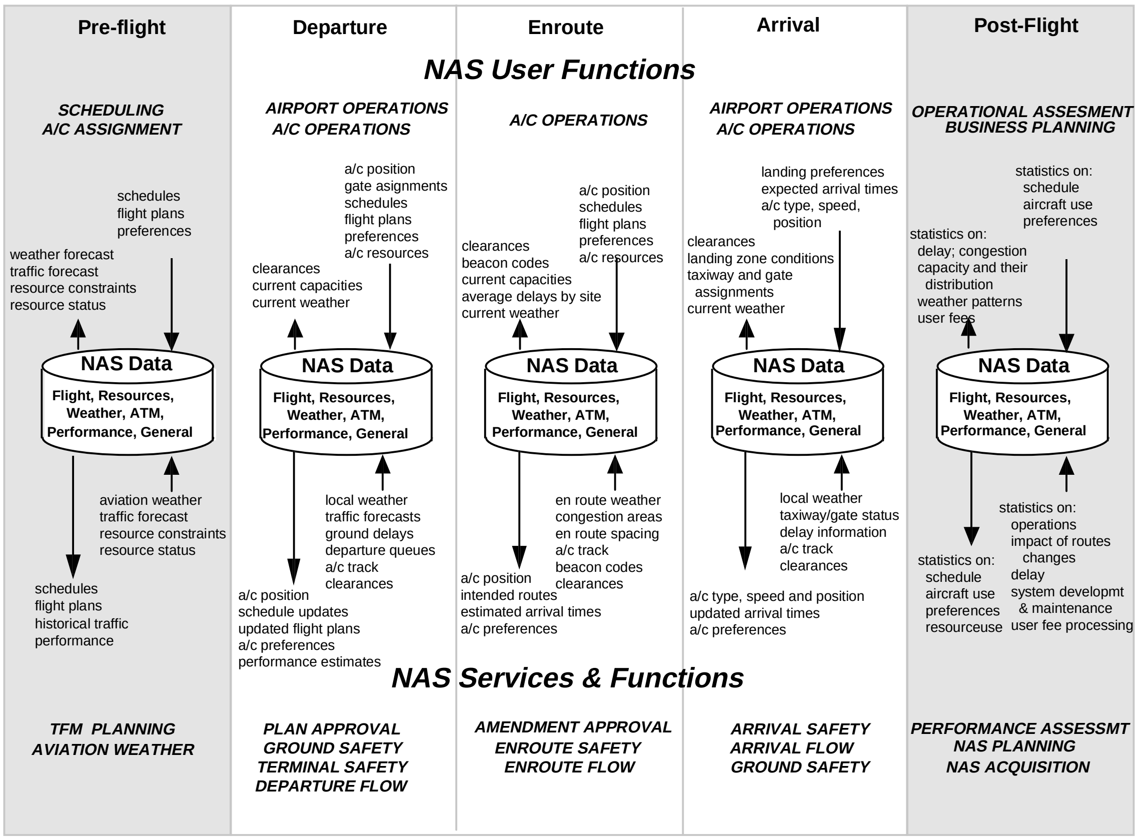

Figure 2-1. Data Support for a Phase of Flight View

At present, flight data are represented uniquely in each domain and in each application system. This document describes a flight data model that could become a basis for a common representation of flight data for new applications using flight data. It could also be applied to systems, such as ETMS and others, that manage flight data and that are to be re-designed in the near future. Based on the analysis of TFM data (see Section 1) the flight related data can be sorted into the generic data categories shown in Table 2-1. Additions to the initial data models are required to span the full set of data categories shown.

Objectives. Before describing the details of the flight data model design, the key term for the model, "flight", needs to be defined. In ATM terms, a "flight" is an

aircraft departure from an airport or other 'takeoff' zone followed by its arrival at another airport or equivalent landing zone, with no intermediate touchdown points in-between. By contrast, a "flight itinerary" is a set of concatenated "flights," in which the departure airport for the next "flight" is the arrival airport of the previous "flight," except for the first leg of the itinerary. A flight is

Table 2-1. Flight Data Categorization

Category Number	Category Name	Description
1.	Flight (Demand)	The data required by NAS users and operators to describe, manage, and control the safe movement of aircraft in the NAS. Much of this data is associated with one flight where a flight normally includes one take-off and one landing.
1.1.	Flight Itinerary	The description of an aircraft operation involving multiple takeoffs and landings.
1.2.	Flight Operator	The data describing the person(s) or organization(s) responsible for the operation of a flight.
1.3.	Flight Identification	The data needed to uniquely identify a flight.
1.4.	Flight as Planned	The data needed to describe a flight to be made at some future time. May include alternatives and preference from many sources (e.g., users, ATC, TFM, automation, policy and procedures).
1.4.1.	Route Preference	The description of preferred route from take-off to landing.
1.4.2.	Departure Preferences	The description of preflight and departure operational preferences.
1.4.3.	Arrival Preferences	The description of arrival operational preferences.
1.4.4.	En Route Preferences	The description of en route operational preferences.
1.4.5.	Descent Preferences	The description of descent profile preferences.
1.4.6.	Diversion Preferences	The description of alternative airport preferences for IFR flights.
1.4.7.	Flight Plan Amendments	The record of changes made to an approved flight plan.

1.5.	**Flight as Flown**	**The description of the flight as flown including measured parameters and actual times of events.**
1.5.1.	**Flight position reports**	**The measured or estimated position of an aircraft from a surveillance system.**
1.5.2.	**Flight events**	**The description of activity that is normally associated with a specific time.**

Table 2-1. (concluded)

Category Number	Category Name	Description
1.5.2.1.	Flight Path events	The description and timing of an event associated with aircraft movements, (e.g., wheels up).
1.5.2.2.	ATM control events	The description and timing of an air traffic management action taken that affects flights.
1.5.2.2.1.	Single flight control	The description and timing of an air traffic management action (e.g., controller action) that affects a single flight.
1.5.2.2.2.	Traffic management strategy	The description and timing of an air traffic management action (e.g., TFM constraint) that affects multiple flights.
1.5.3.	Flight status	The description of dynamic flight parameters or state variables (e.g., velocity, altitude, position, assigned beacon code, etc.).
1.5.4.	User Fees	The data needed to assign user charges (if implemented).
1.6.	Flight as Forecast	The data describing predicted flight parameters and predicted times for events.
1.6.1.	Flight position predicted	The data describing the predicted location of an aircraft.
1.6.2.	Flight event predicted	The data describing the predicted time of a flight event.
1.7.	Aircraft	The data describing the aircraft used in a flight.
1.7.1.	Aircraft description	The static data identifying and describing an aircraft.
1.7.2.	Aircraft status	The dynamic data (e.g., fuel on board) affecting how a flight is flown.
1.7.3.	Aircraft equipage	The description of equipment (e.g., navigation systems) that affect how and where a flight is flown

defined by a flight identifier, its departure airport, and its departure time. In terms of the data model, “flight” represents an abstract information domain which includes states, events, processes, activities, and data related to a flight designated by a flight plan.

The flight data model is designed to achieve two primary goals. These are:

- **To propose a common view of flight data used by applications that includes common names, definitions, and constraining relationships; a fundamental step toward data sharing.**
- **To model information about a flight, including its events and corresponding status information, so that real-time flight data as well as post-event historical data can be managed.**

Flight information comes from numerous sources: pilots, airline operators, service operators, ATM systems (e.g., the Host), and the aircraft itself via Data Link or Global Positioning Systems (GPSs). Real-time flight information is naturally needed by service providers and flight operators to safely and efficiently manage an individual flight as it is being conducted. Flight data is part of the post-event historical data needed by analysts to understand the behavior and dynamics of the NAS, as well as how these behaviors and dynamics are influenced by capacity and procedural changes.

Approach. Since the objective of the flight data model is to capture the behavior of "flights", a structured database design methodology was used to represent the flight objects (entities) and their relationships. This methodology, combining an information model, a state model, and process model, requires three steps that define:

- **The objects that make up a flight model, including data entities, data attributes, and relationships among them.**
- **The behavior of objects, in which each object and relationship may have a life cycle which is a pattern of behavior. For example, a flight plan goes from initial submitted status ('filed') to 'cleared,' and then to 'active,' and finally to 'terminated' (by cancellation or termination at the end of a flight).**
- **The activities and events involved in each state of an object. For example, during the active state of a flight plan, amendments can be added to change its content.**

Implementation. There are two distinct concepts in the flight data, as modeled. First, there is a set of data entities and attributes that deals with the identification of a flight and describes the flight plans (intent) for the flight. Second, there is a set of data entities and attributes that describes the flight as flown, (e.g., assigned aircraft and pilot, tracked position, interactions with the ATC system).

Figure 2-2 illustrates a high level view of a common data model for flight related data. The shaded components are the core of the fight data model. The other components represent placeholders for entities or collections of data that may be modeled in detail as part of other data modeling efforts. For example, extensive data

models have been prepared for FAA resources, e.g., airports, runways, as part of upgrading the FAA's Aeronautical Information System (AIS).

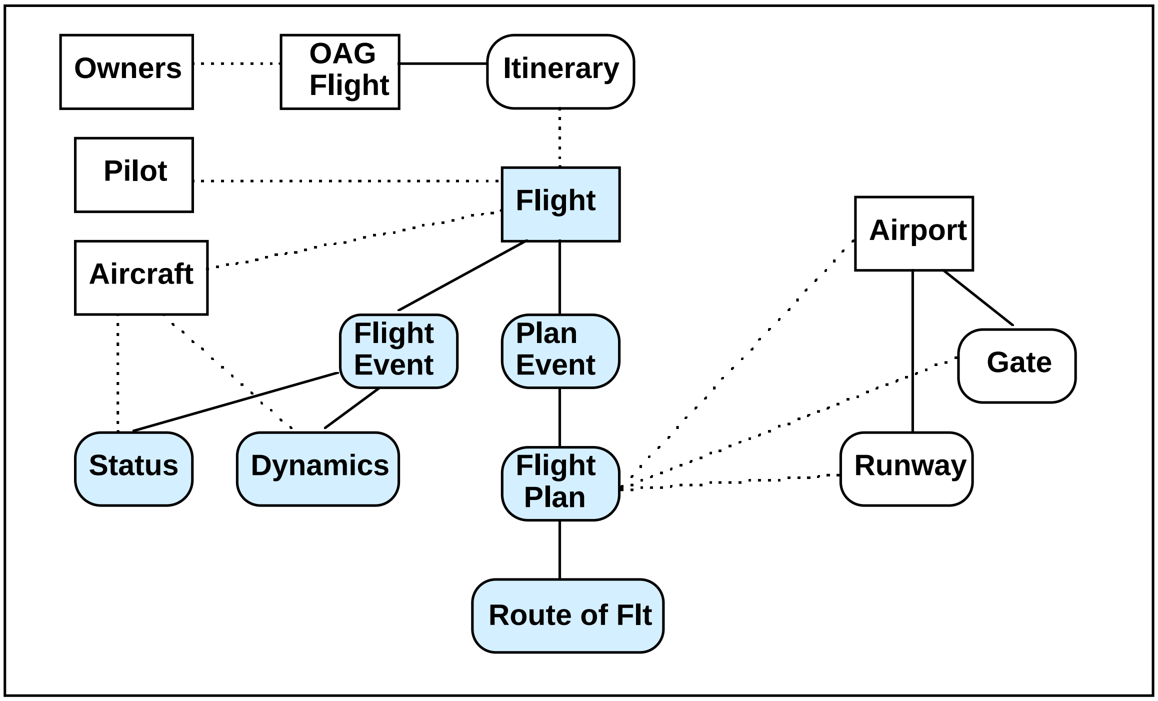

Figure 2-2. Overview of the Flight Data Model

The TFM data model provides a structured representation of the elements of the data standard. The modeling activity used Logic Works' ERwin product, a COTS data modeling tool. ERwin uses the IDEF1X data modeling methodology that was developed for the U.S. Air Force. This methodology is now widely used where large scale, rigorous, enterprise-wide modeling is essential. Figure 2-3 summarizes the IDEF1X data modeling notation.

The ERwin schema of the flight data model in Figures 2-4 and 2-5 depicts the entities, their attributes, and relationships among the entities. For the two concepts mentioned above, the flight data model shows two linked sets of entities for each concept: a flight plan sub-model and a flight as flown sub-model. Each is driven by events: plan events and flight events. Events result in the change or creation of instances of data entities. Event sources are pilots, airline operators, ATM service operators (controllers, and traffic management specialist), ATM systems (e.g., the Host), and aircraft (via data link or GPS). The two sub-models are connected through

a Flight entity that contains a unique system-generated flight_record_id that propagates throughout the flight related data entities. The Flight entities also includes attributes (flight_number, departure_airport, departure_time) that provide alternate keys for identifying a flight.

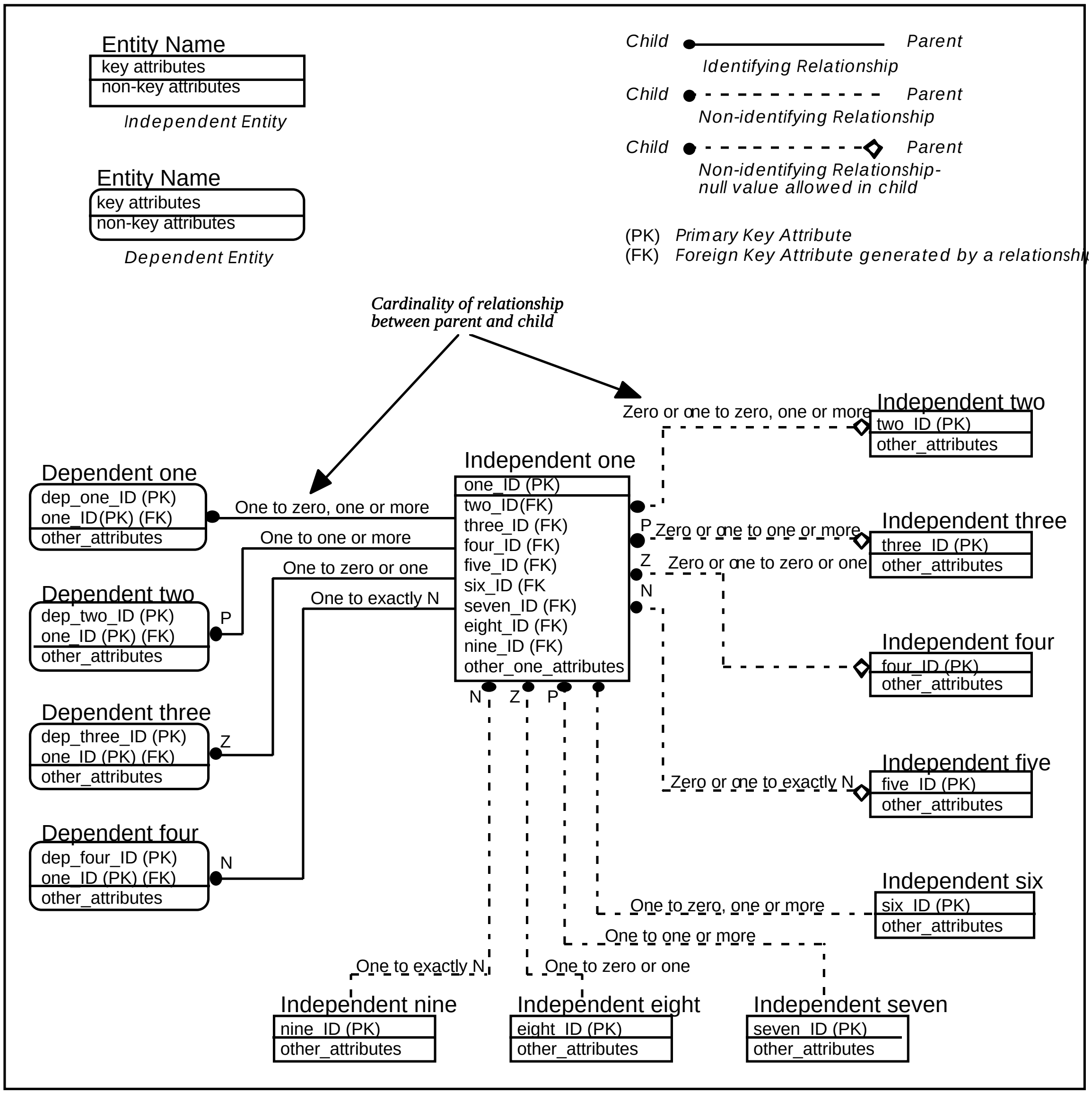

Figure 2-3. Data Modeling Notation Based on IDEF1X Standard

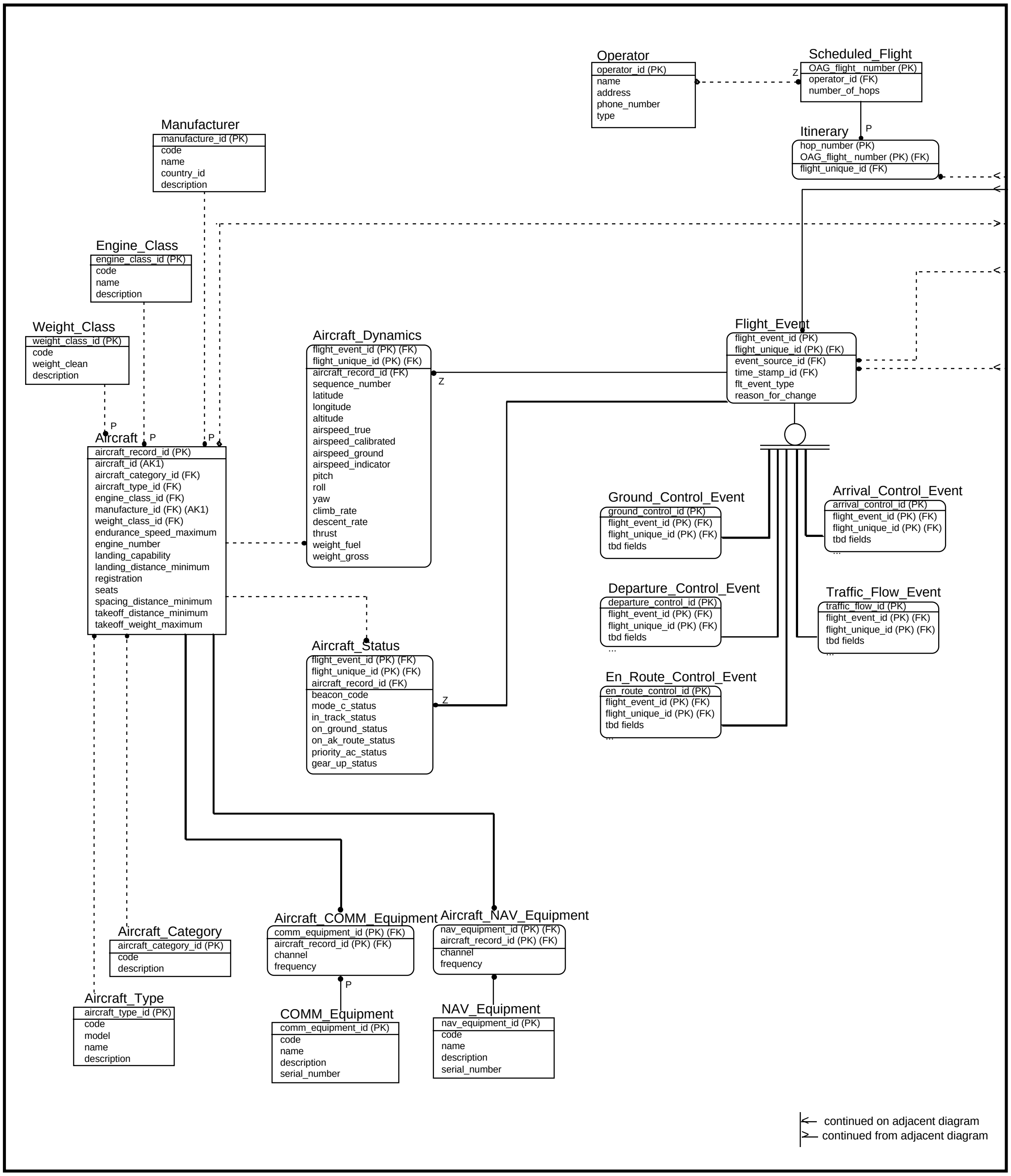

Figure 2-4. Flight Data Model—Flight Data (part 1)

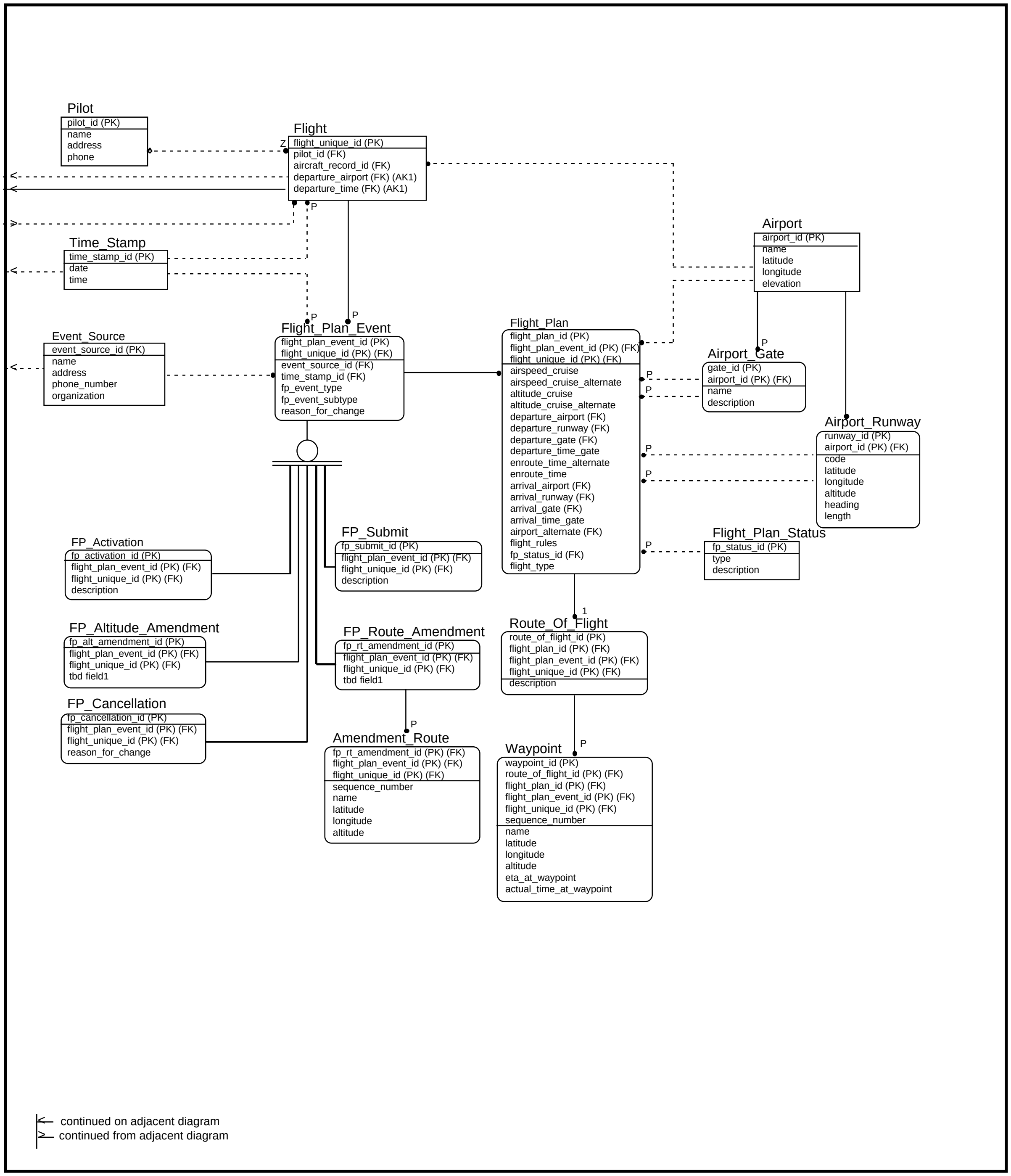

Figure 2-5. Flight Data Model—Flight Intent Data (Part 2)

Section 3

NAS Information Services Engineering Repository

The development of a system engineering repository containing data structures (metadata) and systems engineering data is one of the ways to improve the system engineering process in an organization. Today, even though NAS systems interoperate, have overlapping functional ties, and manage similar data, there is no easy way for a program manager or system developer to 'see' the building blocks (e.g., data, functions, hardware, system software distribution/location) of multiple systems at the same time.

A system engineering repository stores information about multiple systems as described above. It is a basic tool to manage a complex, dynamic, multi-system architecture, and has direct implementations for architecture development, cost impacts, and system management.

Metadata is structural information about data. For example, as 'DCA' is the value of the airport identifier for Washington National Airport, the item 'airport identifier' is also data. Rather than being a value for data, as is 'DCA,' it is a value for an item of metadata. The process of system development and maintenance requires metadata, including full descriptions of each metadata item and its relationships with other metadata. Metadata can be viewed as the index to data, i.e., 'airport identifier' is an index to the values of airport identifier for all airports in the NAS. NAS metadata describe data about the operational data, e.g., how the operational data are defined and structured, but are not the operational data themselves.

In current NAS practice, each application system manages its system metadata locally and may or may not use structured methodologies and tools, that are available to manage a metadata repository. As a result, metadata information is usually accessible only internally through an application and is not available for system integration and for building interoperability into NAS systems. The effort to assemble metadata information is then unnecessarily duplicated system by system, and there is not a consistent view of information across NAS systems.

The NAS Architecture can be strengthened, therefore, by including metadata management, in a NAS Information Services Engineering Repository, as part of the NAS Information Architecture as an automated resource that is "used to describe, document, protect, control and access informational representations of an enterprise."[5] Via such a repository, these metadata would be managed and disseminated throughout

---

5 *On Repositories, Briefing,* B. Parker, The MITRE Corporation, McLean, VA, June 1995.

the three levels of the NAS Information Architecture and would be available to NAS users to facilitate data exchange with the FAA and among users as well.

Although a NAS Information Services Engineering Repository would be part of the Technical Architecture, it plays an active role in bridging information management across application systems, which are associated with the Logical Architecture. It can be a key component required to deliver common information services across the NAS such as common data directory services.

Information in an Information Services Engineering Repository contains more than just metadata *per se*. The data can also include information about data systems and components such as:

- **NAS data structures, data definitions and formats, and data relationships**
- **Application system ownership and responsibility**
- **Application system functionality**
- **Data requirements and data associated with system functional requirements**
- **Application system interfaces, including messages and the data they contain**
- **Application system software architecture, including software design and languages**
- **Application system cost, schedules, and budgets**

Figure 3-1 is an initial information services engineering repository design that incorporates the type of information services engineering in the bulleted list above. There are several purposes for a metadata repository in the NAS in addition to its use in system development and maintenance. These purposes are:

- **Management oversight: To enable managers quickly and easily to search an information services engineering repository for answers about the status of the NAS architecture in general and about individual systems in particular**
- **Investment analysis: To provide key inputs about the costs and schedules of various NAS development programs to cost/benefit algorithms and decision support routines and for the development of tradeoffs in agency resource allocations**
- **Operational performance: To support and track the delivery of NAS information to decision makers across NAS facilities**
- **System development: To monitor system development efforts — from requirements tracking, to the development of system interfaces, to field deployment and integration into local facility architectures and to provide traceability from legacy system data to the NIS based data.**

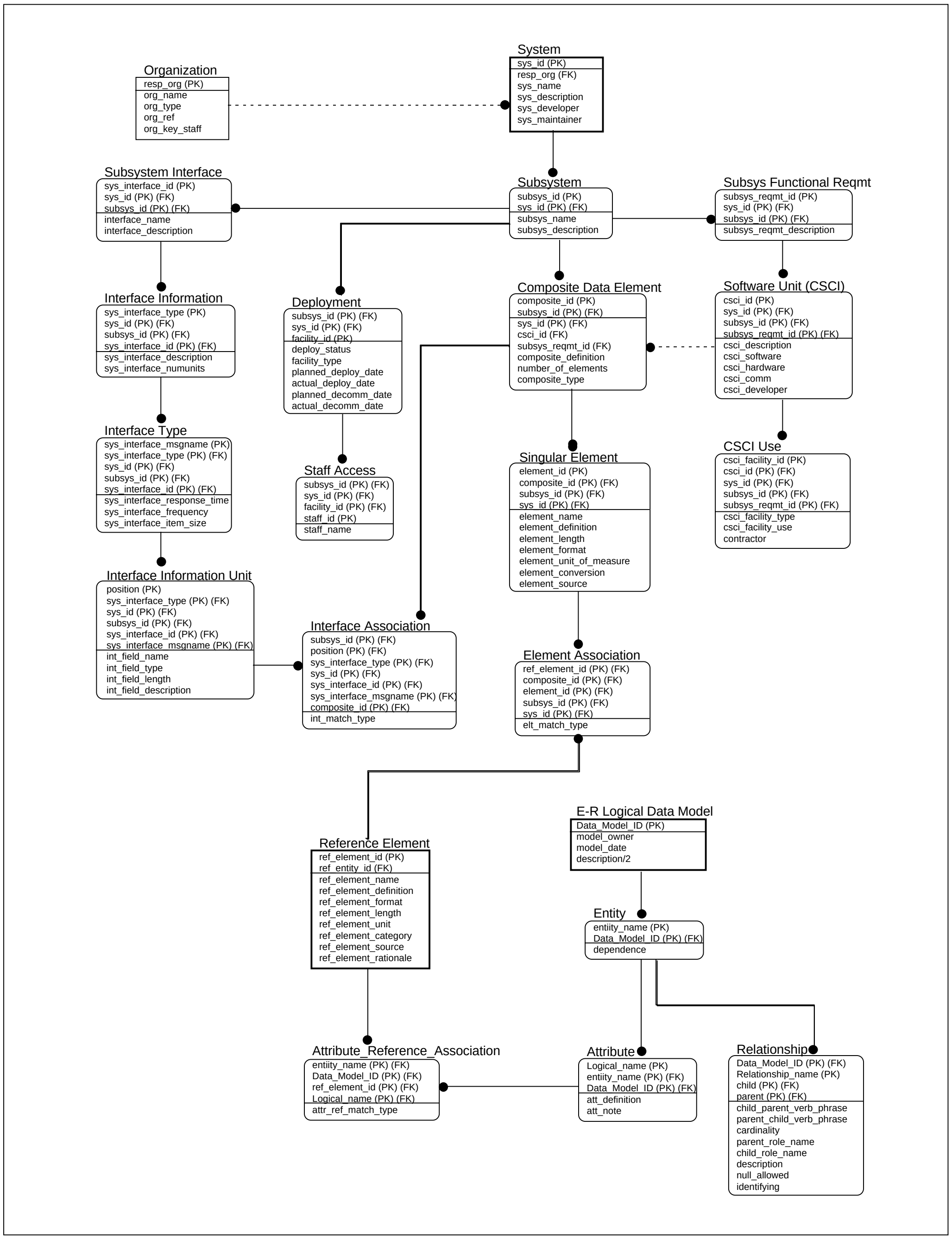

Figure 3-1. NIS System Engineering Repository Data Model

In addition, a repository will maintain traceability between data in legacy systems and in the data models defined for new and re-designed systems. It will also facilitate the sharing and reuse of information about NAS data with the many users of these data, including NAS users and system developers. The latter will use this information for developing new systems and for transitioning legacy systems.

NIS System Engineering Repository Concept of Operations. An operational concept for a system engineering repository is a byproduct of the operational concept for the NAS Information Architecture and is motivated by the need to have answers to the following basic questions about existing and planned NAS data [adapted from Parker]:

- **Who are the users of the data?**
- **What are the data elements and the data architecture of the system?**
- **Where do the data originate and where are they used?**
- **When in the system life cycle are the data needed?**
- **How will the data be needed in the planning process?**

These questions then translate into the following activities that define a NIS system engineering repository operational concept:

- **Manage information about current and planned NAS information resources in a systematic way.**
- **Develop and maintain an automated means (e.g., database and related processes) to manage information about NAS data resources at a NAS-wide level as well as at local (i.e., facility and applications system) levels in the NAS.**
- **Manage the efficient flow of information between data sources and data users (i.e., people and automation).**
- **Manage a recognized, standardized inventory of information 'modules' (e.g., data definitions, data formats, data models) that are immediately available to system developers and NAS users.**

This operational concept can be implemented by developing a NIS System Engineering Repository that consists of the following components:

- **Information needed by and generated by each application system**
- **Functional requirements per application system**
- **Metadata, e.g., data descriptions, data structures, and relationships between data objects**
- **Interface information (e.g., message structure and content)**

- **The source of each type of NAS information (e.g., for weather data, flight schedules, system resources and their capacities)**
- **The locations (e.g., facilities, systems, users) where each type of information is required**
- **Historical versions of the same NAS information to assist in system transitions and traceability among related systems**

Version management and configuration management of the NIS engineering repository will help to maintain a logical link between historical, or legacy, information and current forms of that information. Version management will track iterations of engineering designs during system development. Configuration management will track groups of versioned data objects defined as configurations.[6] [Bernstein]

An engineering repository has many roles to play. When it was first introduced in the industry, it was a fairly passive player in its data management role. Hence, the name 'repository' was applied. Over time, it has become more active in helping to manage information in a system. 'Passive' repositories became 'active' repositories which have now become 'dynamic' repositories. The descriptor refers to the role of the repository in keeping its system-based metadata synchronized with changes in operational data and application system changes that are being made. It both leads and follows these changes under the guidance of a data administrator (DA) and his or her staff.

An engineering repository system architecture consists of three components [adapted from Bernstein]. These are:

- **An information model—specifies the type of information stored in the repository.**
- **A repository engine—manages the repository objects. It is typically a layer of software that is positioned between the repository database residing in a COTS DBMS and the information model.**
- **A repository database—is the technology used to store the repository information.**

---

6 *The Repository: A Modern Vision,* P. Bernstein, Database Programming and Design, Volume 9, Number 12, pp. 28-35, December 1996.

Relationship to Other Architecture Components. The three paragraphs below describe the relationship of the NIS Engineering Repository to other key architecture components.

Operational Concept. A NIS System Engineering Repository has an important role to play across the NAS, especially in tracking the data that originates from outside of FAA boundaries. The repository can be a focal point for data exchange and interoperability across the FAA/NAS user boundary. This is especially so since there is not just one user but many—the scheduled air carriers, general aviation (GA) and freight carriers; the military; international aviation. The repository will track the source and destination of NAS data among organizations, systems, and locations. In this way it will be a key part of implementing data exchange and collaboration in line with what the operational concept envisions.

Infrastructure. Infrastructure is concerned with the hardware that provides processing capability, storage, communications and connectivity, and with software, including operating system software and application system software. The NAS Infrastructure Management System (NIMS) is being designed and built to manage information that contains the status of NAS facilities and equipment. However, its view is mainly location or facility-oriented. A NIS System Engineering Repository will complement NIMS by tracking the infrastructure components associated with each applications system. This information will include the locations at which the applications are operating and the equipment at each location.

Application Architecture. In addition to serving a data dictionary function for NAS data structures, the repository will also complement NIMS by managing functional information about applications systems. In addition to its primary function of managing NAS data structures, a NIS System Engineering Repository will also manage information that decomposes an applications system into its functions and the data associated with each function. It will track the sources of all information required by the application and the destinations receiving information from each system in terms of content, rate, and volume.

The repository will also manage information about the interfaces that an application system has with other systems. This information consists of the set of messages sent by the focus system to its interfacing systems as well as the (atomic) data contained in each message. This information, together with information about application deployment mentioned above, will provide analysts with the capability to tune the system in response to changing performance requirements that include both data timeliness and content.

Bibliography

Bernstein, P., December 1996, "The Repository: A Modern Vision," Database Programming and Design, Volume 9, Number 12, pp.28-35.

Cherdak, S. J. B., et al., September 1996, *Traffic Flow Management Research and Development Activities,* MP96W200, The MITRE Corporation, McLean, VA.

"Digital Media Management: A White Paper," Sun Microsystems, 13 February 1997; found at http://www.sum.com/media/presentations/dmmwp/whitepaper.htm

"Engineering Information Systems and the IEEE Standard 1220-1994," National Council on Systems Engineering (NCOSE) Conference, July 1996, Boston, MA.

Federal Aviation Administration, November 1993, *The Future Traffic Flow Management System: Volume II, Operational Description,* U.S. Department of Transportation, Washington, DC.

Federal Aviation Administration, June 1996a, *A Concept of Operations for the National Airspace System in 2005, Revision 1.3,* U.S. Department of Transportation, Washington, DC.

Federal Aviation Administration, September 1996b, *TFM R&D Project Summaries,* U.S. Department of Transportation, Washington, DC.

Federal Aviation Administration, October 1996c, *National Airspace System Architecture, Version 2.0,* U.S. Department of Transportation, Washington, DC.

General Accounting Office, February 1997, *Air Traffic Control: Complete and Enforced Architecture Needed for FAA Systems Modernization*, Accounting and Information Management Division, GAO/AIMD-97-30.

Hermes, M. and Dunham, S., 28 May 1997, "Preliminary Draft: Initial Operational Concept for Traffic Flow Management Information Exchange," Letter F067-L-005, 1297013F, The MITRE Corporation, McLean, VA.

Joint Technical Architecture (JTA), 22 August 1996, Department of Defense (DOD), Version 1.0, p. 2-3.

Karangelen, Hoang, and Howell, July 1996, "Top Level Requirements for an Advanced Systems Engineering Information Repository," National Council on Systems Engineering (NCOSE) Conference, Boston, MA.

Kyungpook, K., *The Essential Distributed Objects Survival Guide*, (see http address to complete)

Masters, M., Dall, A., Monaco, M., August 1996, *Objectizing a Relational Database: An Extensible Approach to Many-to-Many Relationships*, The MITRE Corporation, Presented at the Department of Defense Database Colloquium '96, San Diego, CA.

Moriarity, T., March 1990, *Are You Ready for a Repository?*, Database Programming and Design, Volume 3, Number 3, p. 61-71.

Parker, B., June 1995, "On Repositories," Briefing, The MITRE Corporation, McLean, VA.

Renner, Dr. S., Scarano, J., August 1996, *Migrating Legacy Applications to a Shared Data Environment*, The MITRE Corporation, Presented at the Department of Defense Database Colloquium '96, San Diego, CA.

Rennhackkamp, M., "Building a DBA Repository System," January 1996, DBMS Magazine, Volume 9, Number 1, pp. 59-64.

Rosenthal, Dr. A., Ceruti, Dr. M., August 1996, *Toward Data Administration in the Large*, The MITRE Corporation and NCCOSC, respectively, Presented at the Department of Defense Database Colloquium '96, San Diego, CA.

RTCA, Inc., November 1996, *Minimum Aviation System Performance Standard for Air Traffic Management - Aeronautical Operational Control Ground-Ground Information Exchange*, Washington, DC.

Glossary

AIS	**Aeronautical Information System**
AOC	**Aeronautical Operational Control**
ASD	**(FAA) System Development & Program Evaluation (Organization)**
ATM	**Air Traffic Management**
CAASD	**Center for Advanced Aviation System Development**
CDM	**Collaborative Decision Making**
COTS	**Commercial-off-the-Shelf**
CTAS	**Center TRACON Automation System**
DBMS	**Database Management System**
DOTS	**Dynamic Ocean Tracking System**
DSS	**Decision Support System**
ETMS	**Enhanced Traffic Management System**
FAST	**Final Approach Spacing Tool**
FIO	**Flight Information Object**
FMS	**Flight Management System**
FSM	**Flight Schedule Monitor**
GA	**General Aviation**
GAO	**General Accounting Office**
GDP	**Ground Delay Program**
GIS	**Geographical Information System**
GPS	**Global Positioning System**
GUI	**Graphical User Interface**
HCI	**Human Computer Interface**
HCS	**Host Computer System**
ICAO	**International Civil Aviation Organization**
IPT	**Integrated Product Team**
LIS	**Local Information Service**
MAMS	**Military Airspace Management System**

NAS	**National Airspace System**
NAVAID	**Navigational Aid**
NCIS	**NAS-level Common Information Service**
NIAC	**NAS Information Architecture Committee**
NIMS	**NAS Infrastructure Management System**
NIS	**NAS-wide Information Service**
NIST	**National Institute of Standards and Technology**
NOTAM	**Notices to Airmen**
OAG	**Official Airlines Guide**
OMB	**Office of Management and Budget**
PIREP	**Pilot Report**
SAMS	**Special Use Airspace Management System**
SIS	**System-level Information Service**
SMA	**Surface Movement Advisor**
SQL	**Structured Query Language**
STARS	**Standard Terminal Automation Replacement System**
SUA	**Special Use Airspace**
TCDWG	**TFM Common Data Working Group**
TFM	**Traffic Flow Management**
TFM ART	**Traffic Flow Management Architecture and Requirements Team**
TMA	**Traffic Management Advisor**
TMC	**Traffic Management Coordinator**
TRACON	**Terminal Radar Control**
TRM	**Technical Reference Model**
URET	**User Request Evaluation Tool**